Cycle 1 Prototype Demonstration

Larry O'Brien

Bruce Harris

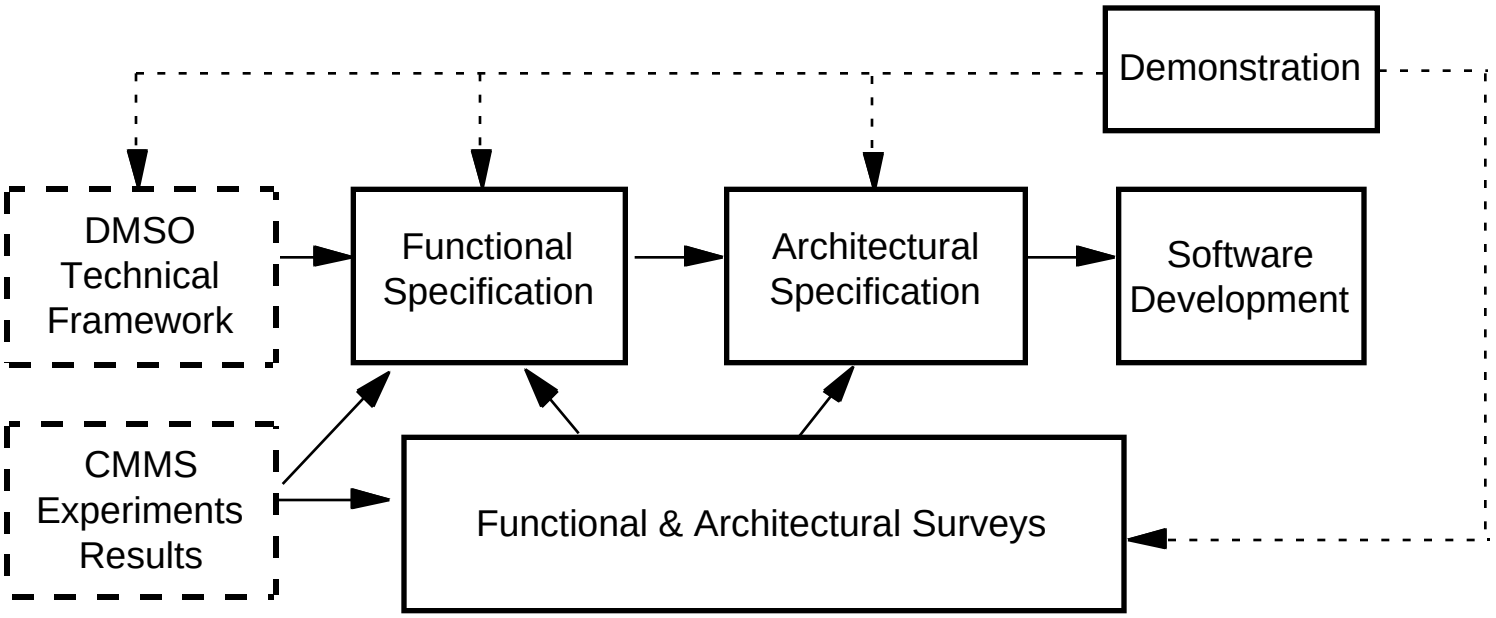
Cynthia Tuttle

Monica Gross

CYCLE 1 DEMONSTRATION

- **Objectives:**
  - Demonstrate all major CMMS functions (except *Create*)
  - Use realistic mission thread
  - Describe lessons learned
- **Based on Technical Framework 0.1.6.2**

PROTOTYPE DEVELOPMENT PROCESS



CYCLE 1 FUNCTIONAL REQUIREMENTS

- Create
- Register
- Convert
- Integrate
- Manage
- Release
- Locate
- Extract
- Evaluate

CREATE

Create Capture of real world information about one or more Missions in the form of a Model for eventual inclusion in CMMS.

- **Goal: Minimize creation of knowledge**
- **Obtain/use knowledge from 4 existing programs**
  - JSIMS
  - WARSIM
  - NASM
  - JWARS
- **Obtain/create knowledge to support integrated mission thread.**

REGISTER

Register	Submission of one or more models for actual inclusion in CMMS including source, format, and content checking with deficiency correction as appropriate.

PRODUCER:

- **Registers new model (e.g., JSIMS, WARSIM, NASM, JWARS) or updates an existing model**
- **Provides summary information on model**
- **Indicates privileges to be granted to users**
- **Describes location of model (model loaded)**

CONVERT

Convert Transformation of a Model which has been Registered from its native form to a standard form required by CMMS including extraction of semantic and syntactic elements**.**

- **Convert data for 4 programs**
  - JSIMS
  - WARSIM
  - NASM
  - JWARS

INTEGRATE

Integrate The act of combining, normalizing, storing, indexing, and in general migrating Registered Models in CMMS standard form to a higher level of structural maturity and semantic enforcement within a unified database

- **Perform relationship checks**
- **Link data to items in data dictionary**
- **Link new data to data already in data base**

LOCATE

Locate The use of on-line browsing tools, automated searches, and retrieval queries to identify Models of interest.

- **Fully Structured Views (FSVs)**
  - Provide Information Interaction FSV
- **Provide Following On-Line Reports**
  - Organization Mission Tasks
  - Task Data
  - Model Semantic Assignments
  - Model Data Quality Errors
  - Model Relationship Errors

EXTRACT

Extract The use of application programming interfaces and automated data retrieval services to gather, format, package, and delivery CMMS Models to simulation developers.

- **Provide on-line queries**
- **Provide capabilities for downloading data displayed in queries and reports**

EVALUATE

Evaluate Determination of the suitability of a CMMS Model for specific end-use by a simulation developer**.**

- **Provide user with on-line interface to update VV&A pedigrees (model version, entity, action, reference)**

MANAGE

Manage The provision of configuration management, version control, change traceability, data storage, and resource allocation for Models in CMMS.

- **Configuration Management**
  - All data tied to model and version
- **Traceability**
  - Interface provided for updating reference documents
  - VV&A pedigree updated to reflect reference changes
- **Data Storage**
  - Implemented via a COTS DBMS
  - Data structure consistent with EATI canonical form
- **Installation Instructions ***
  - Describe installation procedures and requirements including resources
- **Event Services**
  - Producers notified of Integration, Consumers notified of model changes *
  - Integration and Release reports provided

* = Not Implemented in Cycle 1

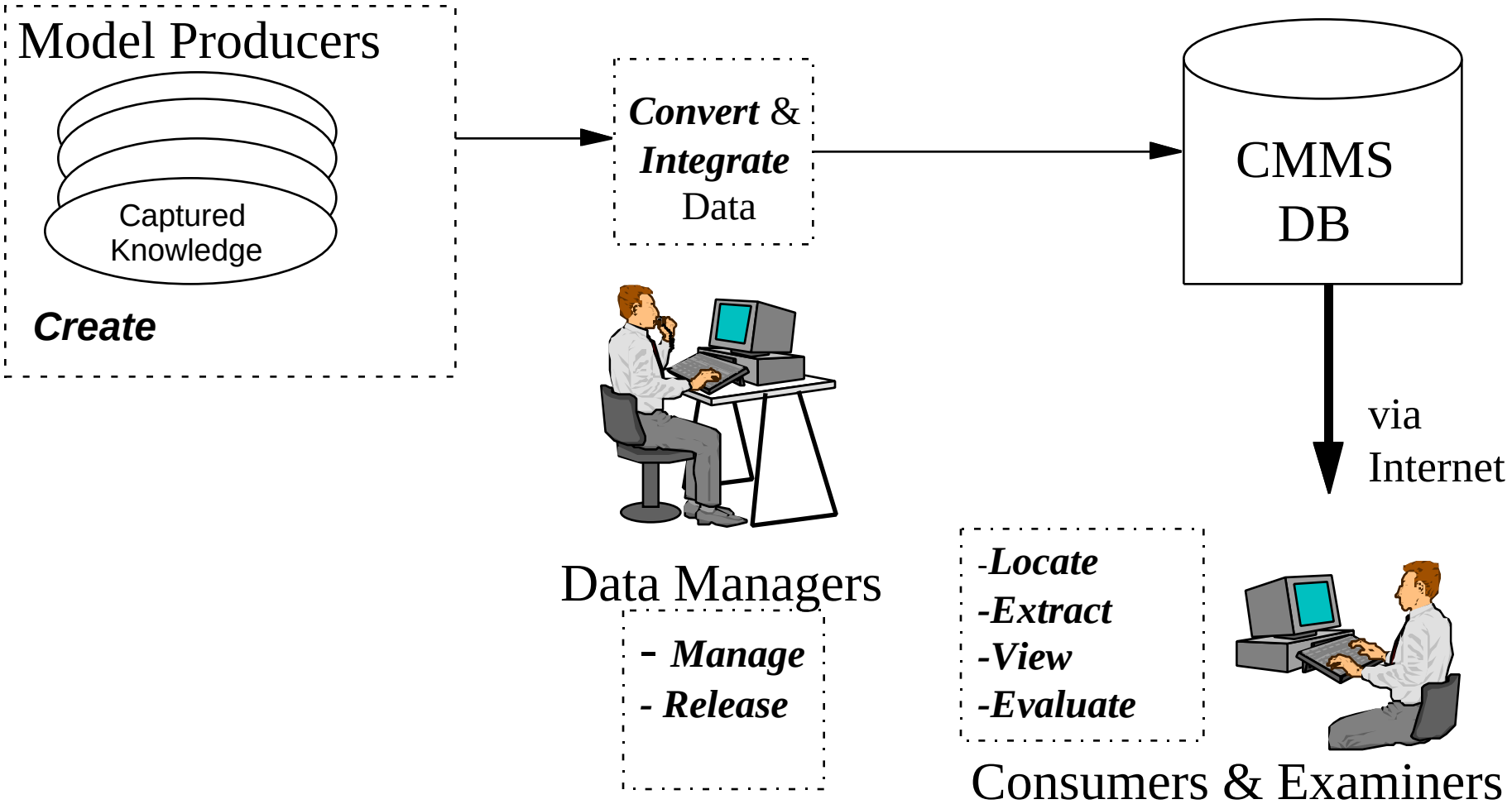
RELEASE

Release The provision of security services, access control, user identification for use and examination of Models.

- **User Identification and Access Control**
  - – Users must be registered and authenticated
  - – Users only allowed to perform functions and access models for which they have been authorized *
- **Security Services**
  - • Cycle 1 implemented on C2 operating system

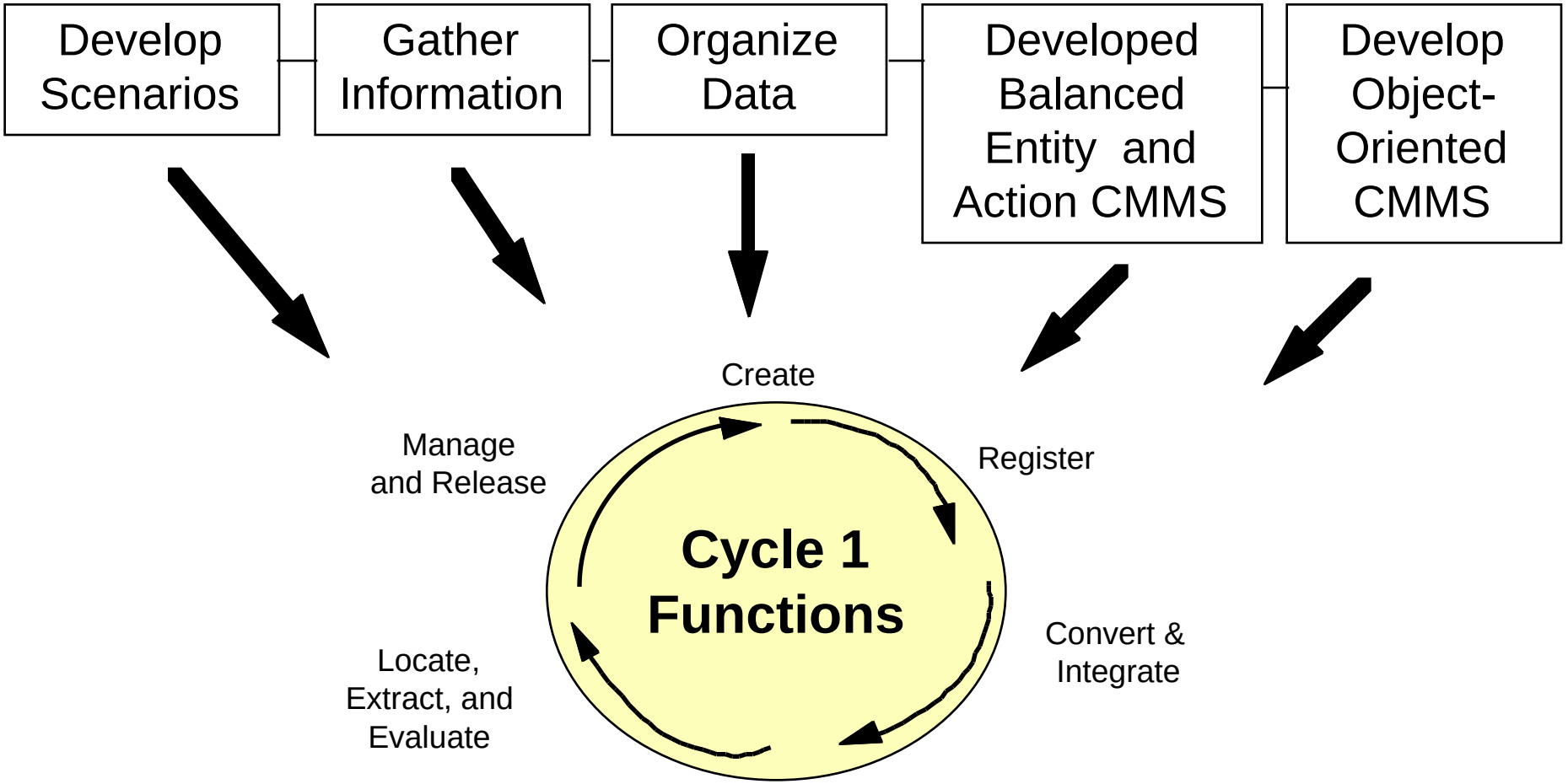
* = Not Implemented in Cycle 1

CMMS OVERVIEW

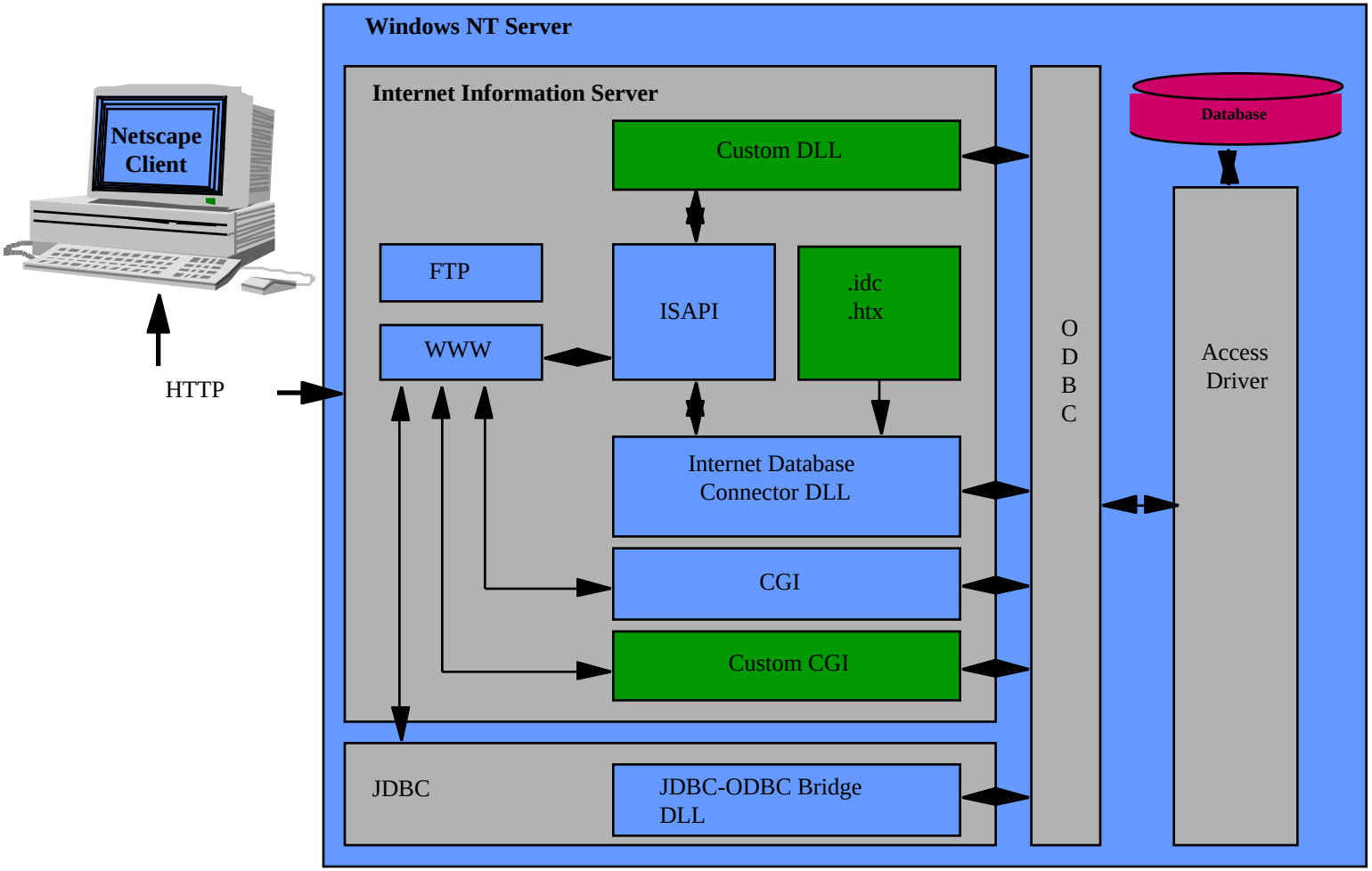


CYCLE 1 FUNCTION & PROCESS

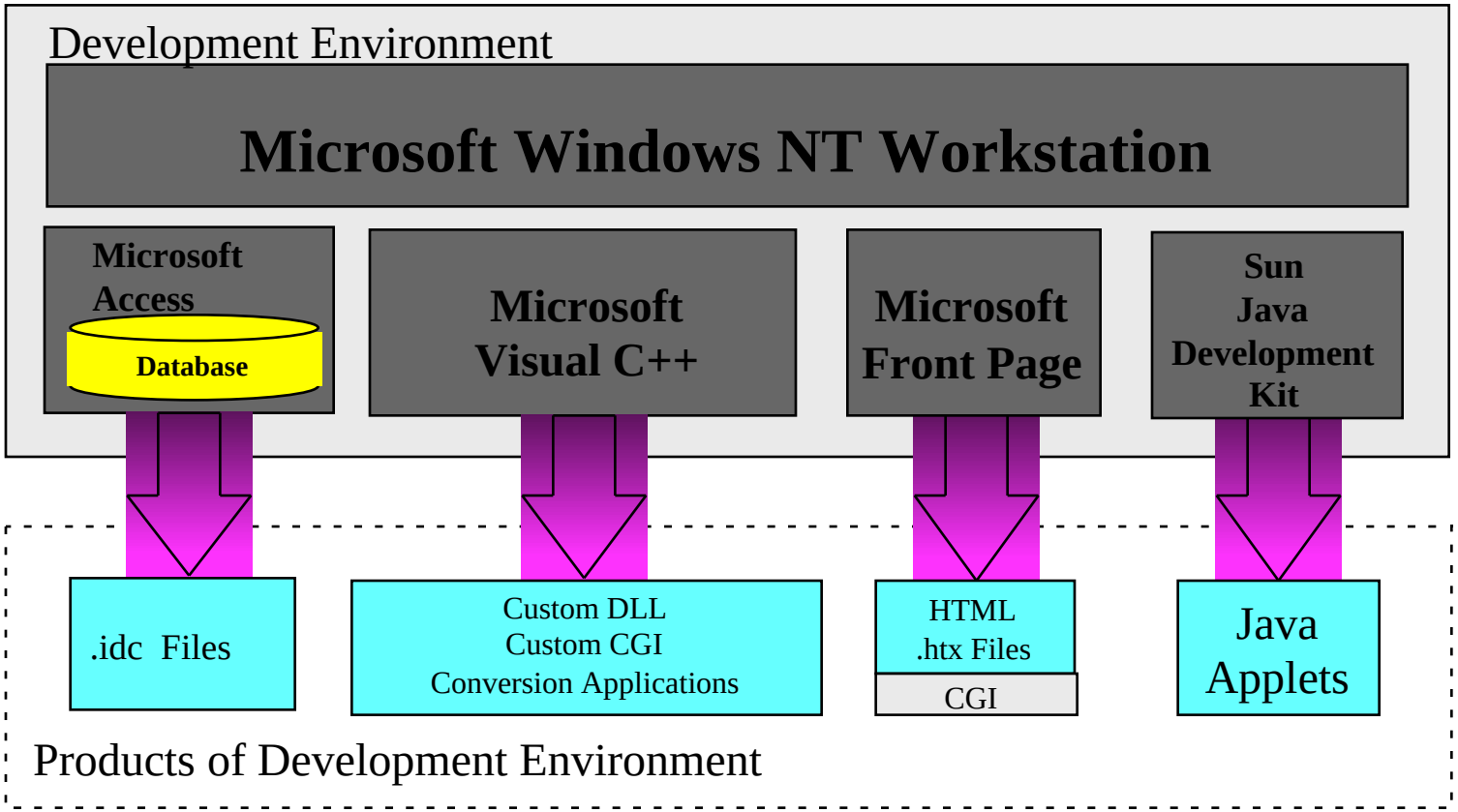
CMMS Development Sequence



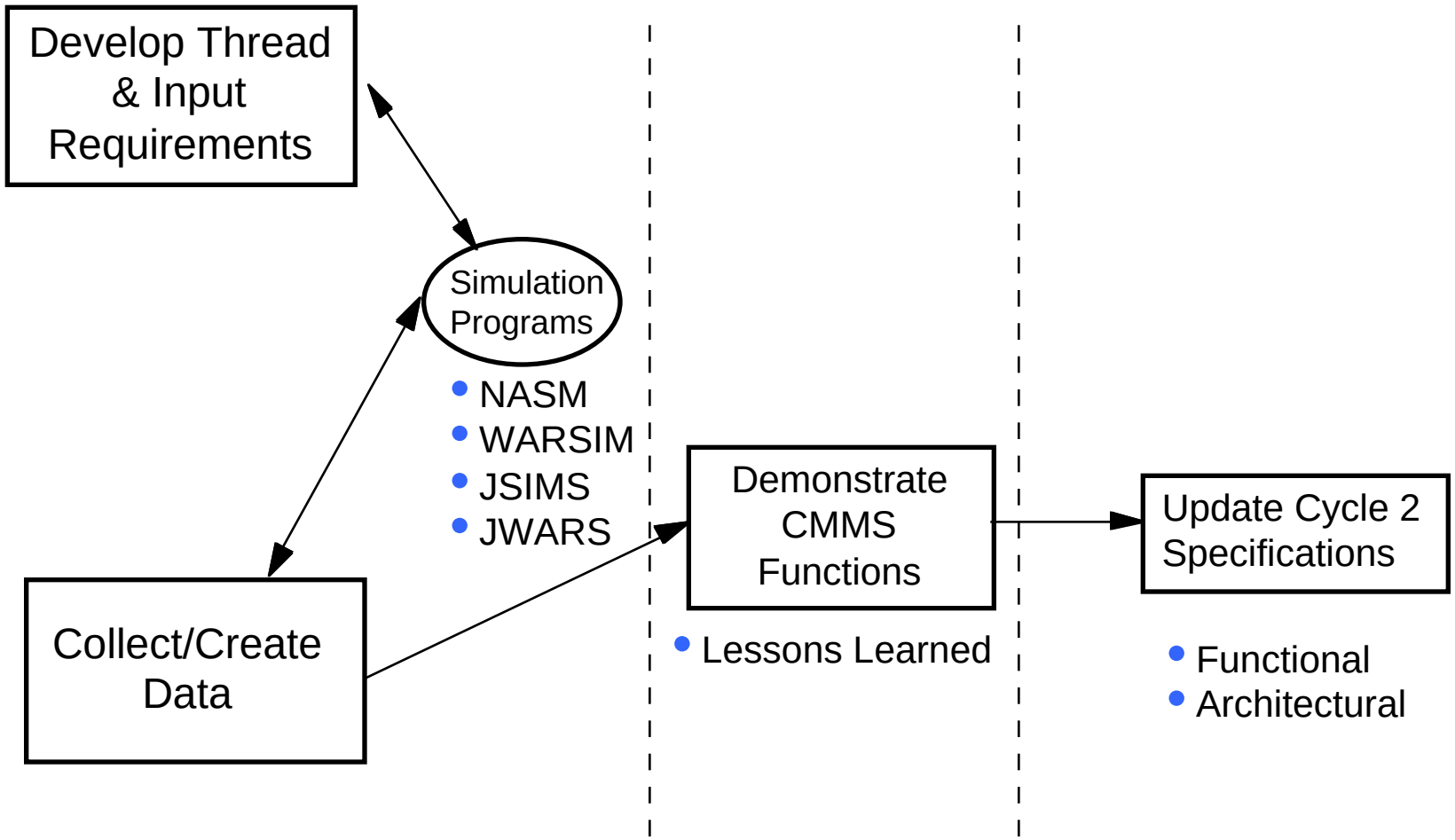
RUNTIME ARCHITECTURE



DEVELOPMENT ARCHITECTURE



DEMO OPERATIONAL CONCEPT



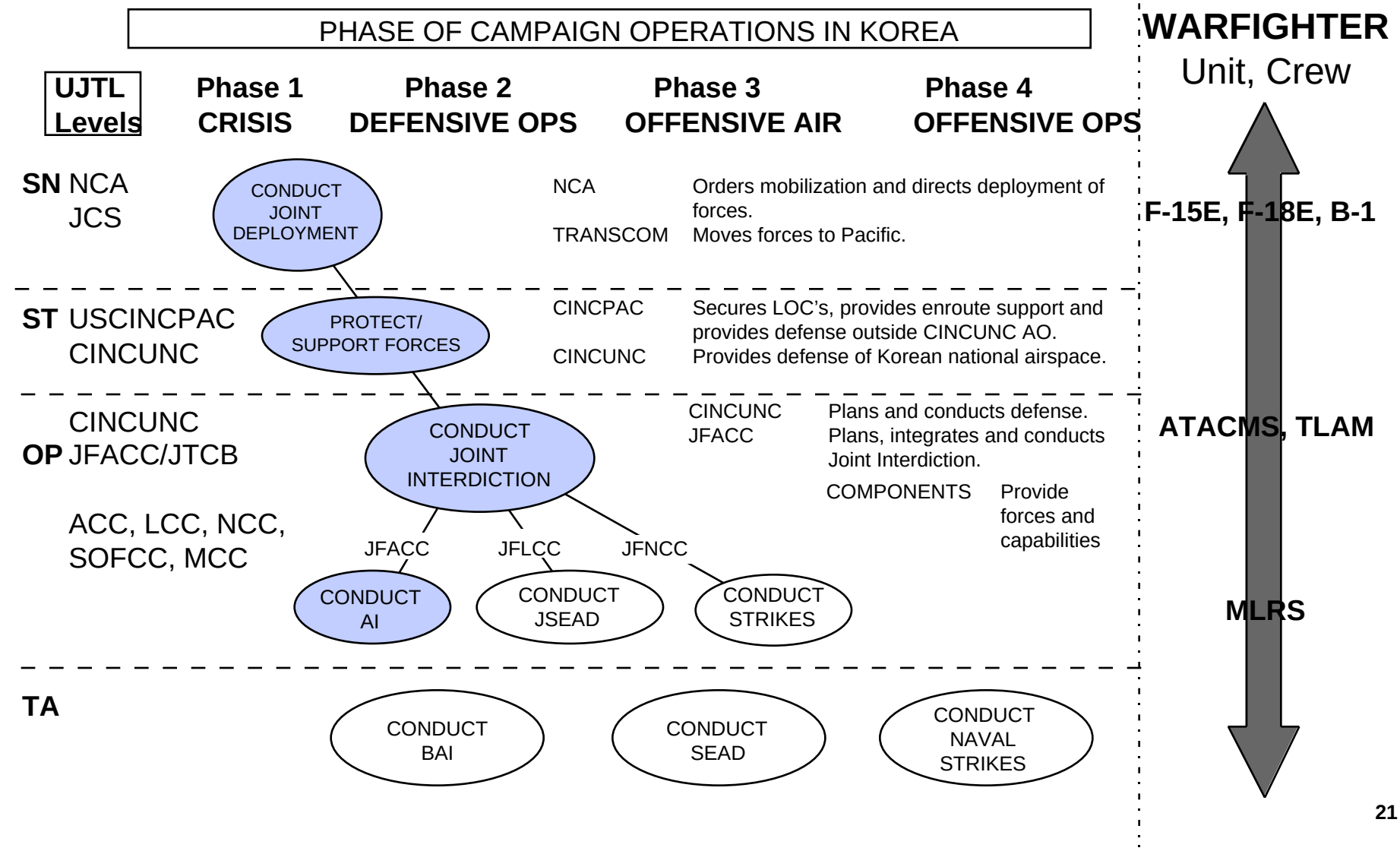
TYPES OF INPUT DATA

Item	Priority	NASM	JSIMS	JWARS	WARSIM
List of Units	High	X	X	X	X
Links to entities in data dictionary	Low				
Unit decomposition	Low	X	X	X	X
List of tasks performed by unit	High	X	X	X	X
Links to verbs in data dictionary	Low				
Links to UJTL (top level tasks)	Low	X	X		
Task decomposition	High	X	X	X	
Inputs and outputs between tasks	Medium	X			
Links to entities in data dictionary	Low				
Assignment of tasks to mission(s)	Low				
Task and unit reference documents	Low	X			

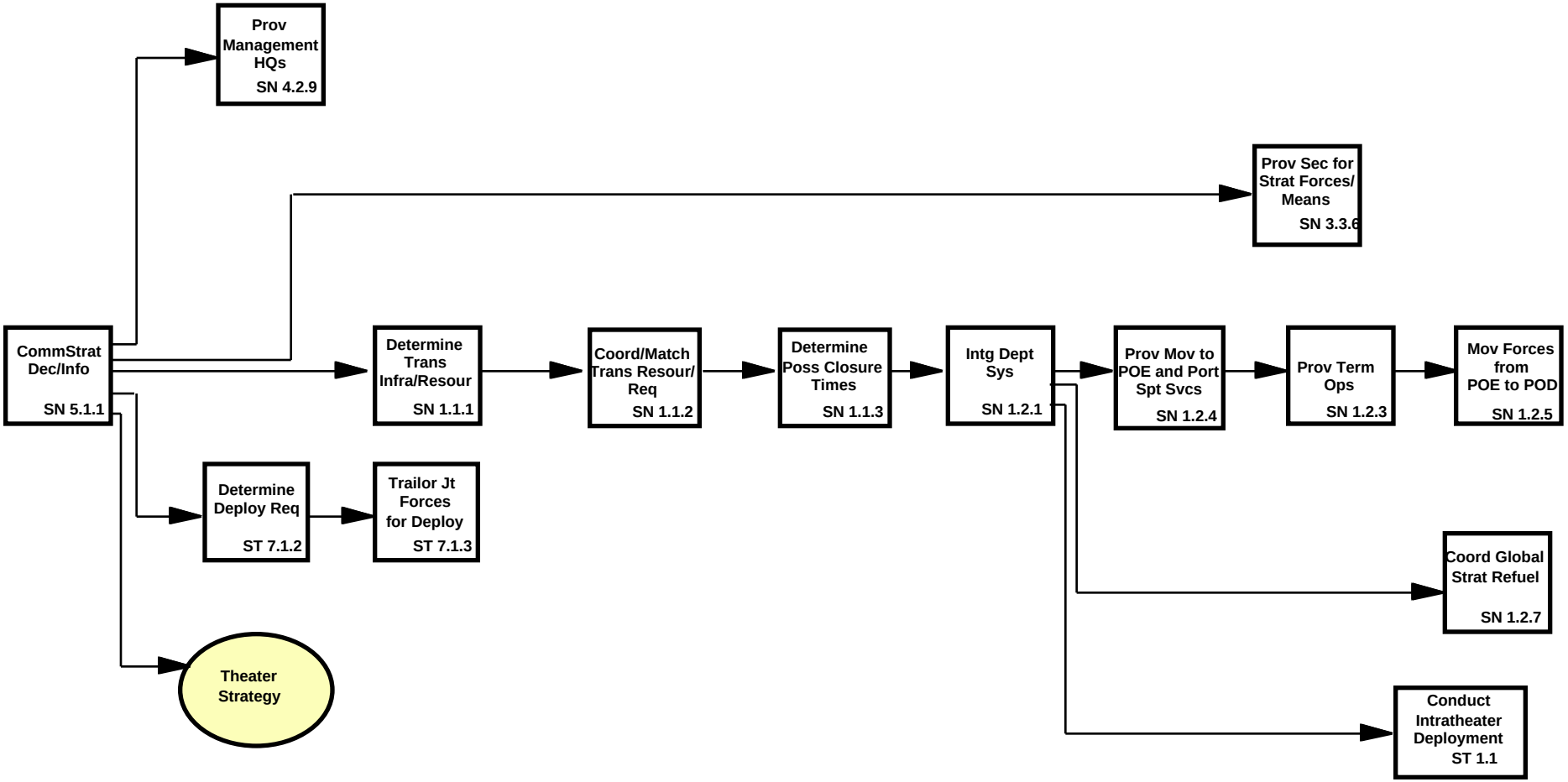
DRC THREAD DATA

- **DRC Created 4 High Level Templates**
- **Each template based on UJTL task and JMETL template concept**
- **Lower level templates linked to simulation program data**
- **Templates**
  - Conduct Joint Deployment (SecDef)
  - Protect and Support Forces (Unified Command)
  - Conduct Joint Interdiction (Sub-unified Command)
  - Conduct Air Interdiction (Air Component Commander)

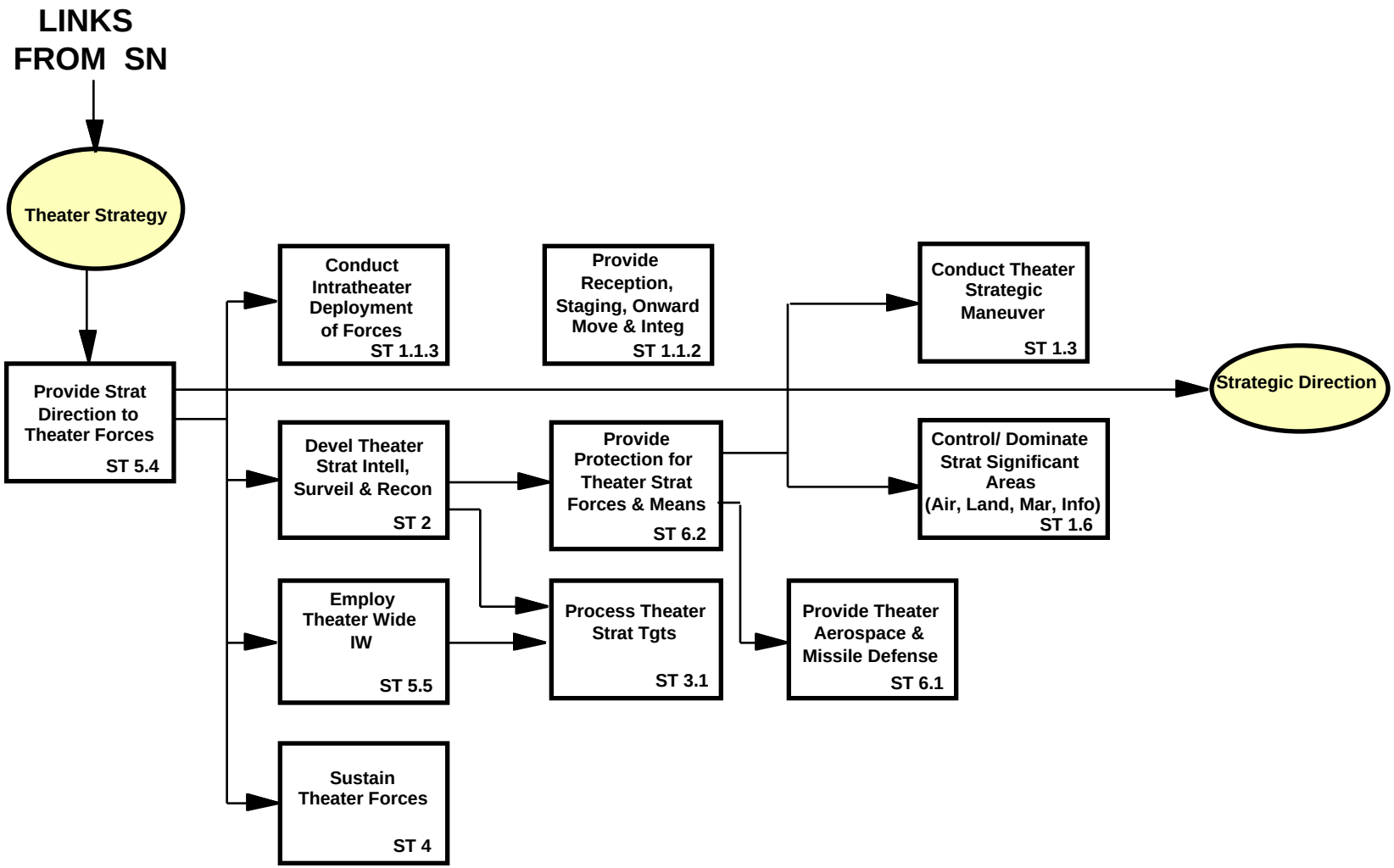
CMMS MISSION THREAD



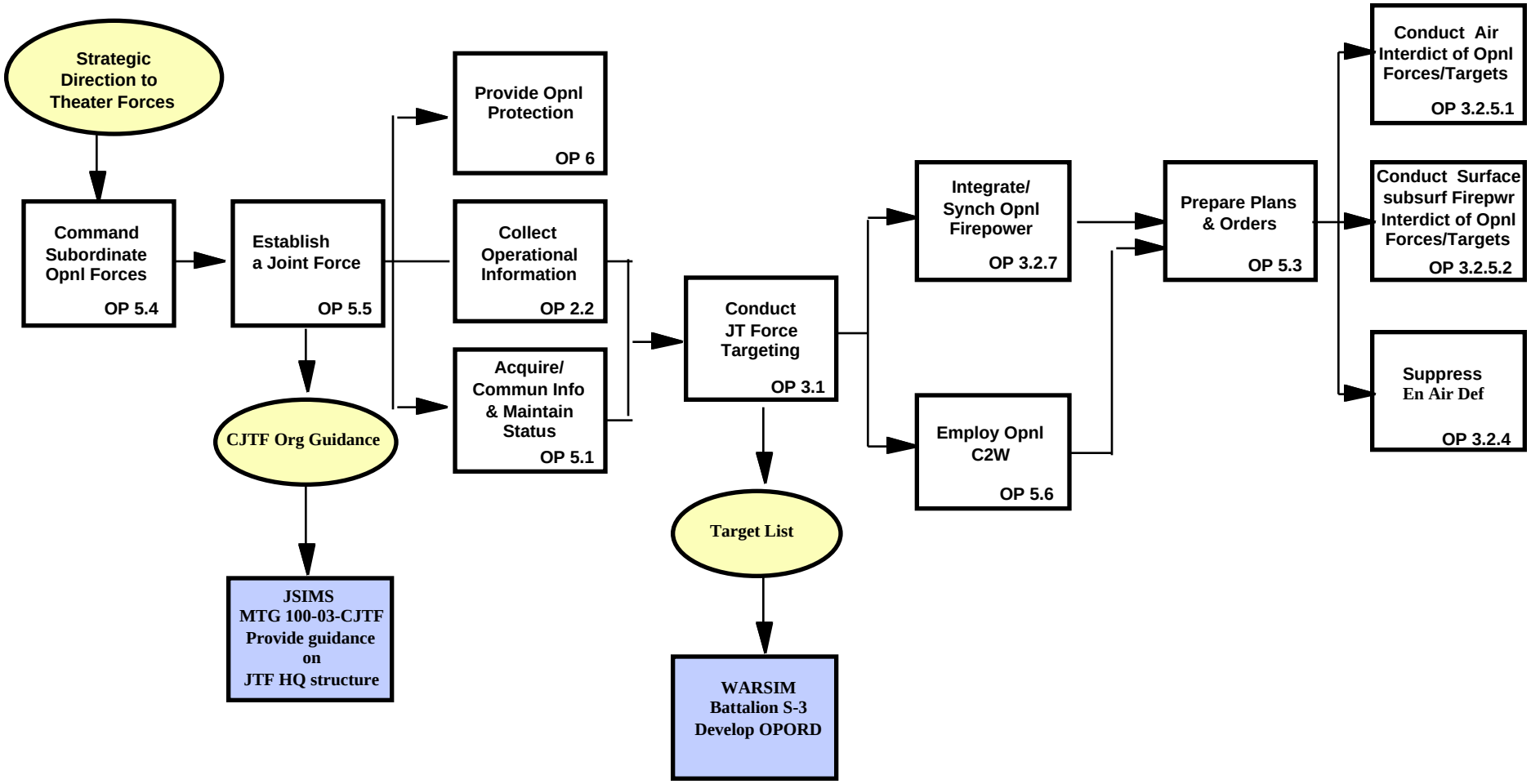
CONDUCT JOINT DEPLOYMENT



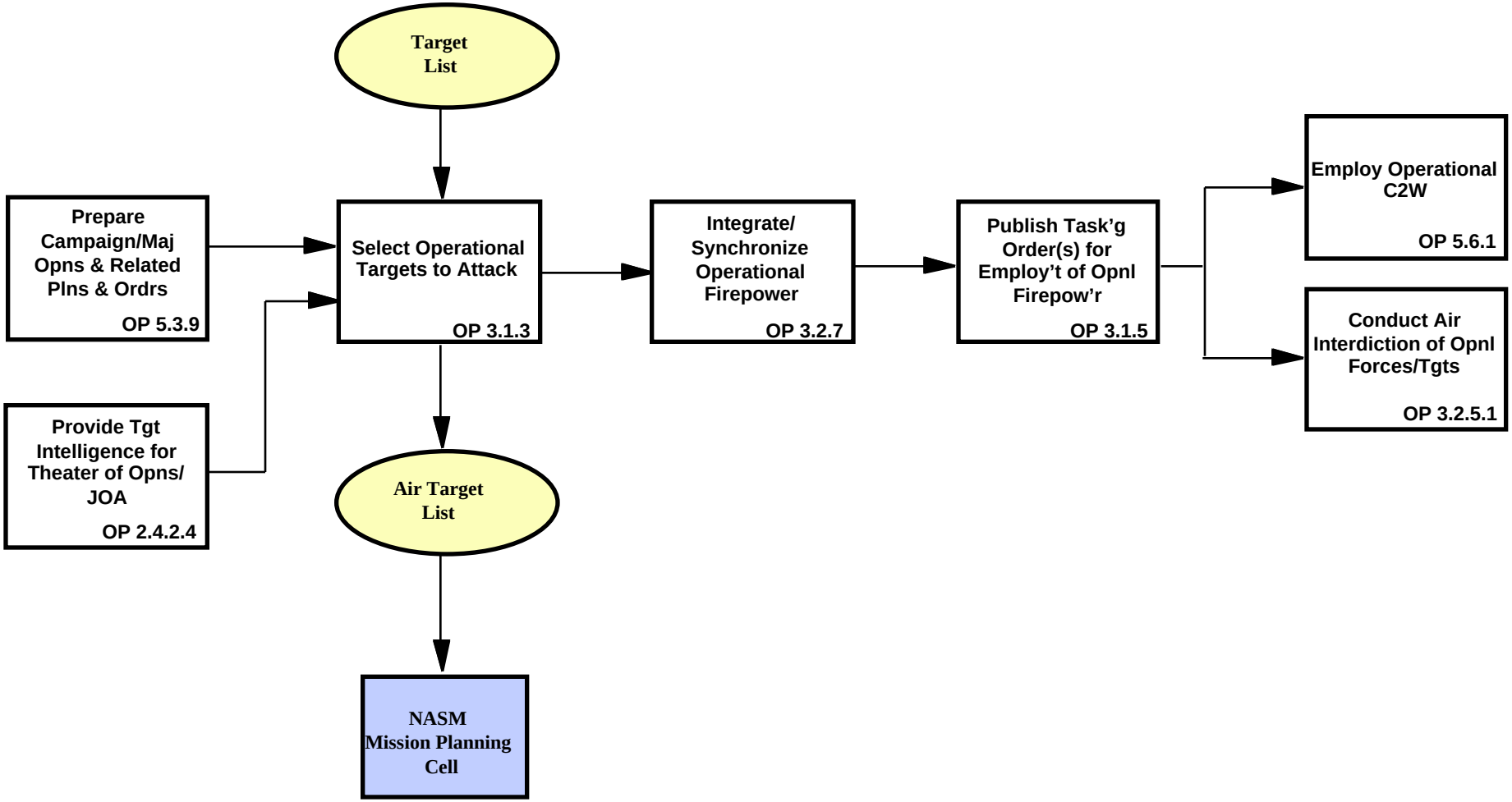
PROTECT AND SUPPORT FORCES



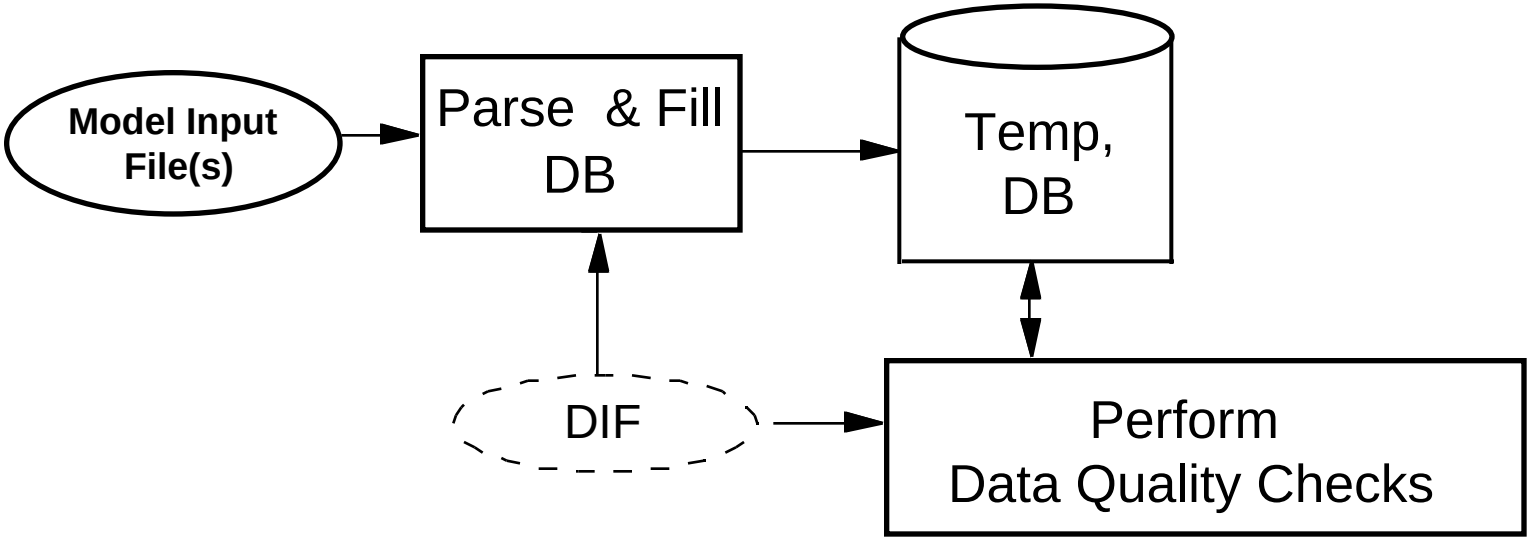
CONDUCT JOINT INTERDICTION



CONDUCT AIR INTERDICTION
(Temporal View)



CONVERSION PROCESS



CYCLE 1 DATA DICTIONARIES

- **Two Dictionaries--Entities & Verbs**
- **Sources**
  - JSIMS Data Dictionary
  - JWARS Data Dictionary
  - Excerpt from Tom Johnson Verb Dictionary
  - NASM RDT Terms (Derived)
  - WARSIM Terms (Derived)
  - DRC Thread Data Terms (Derived)

INTEGRATION PROCESS-SPECIAL CHECKS

- **Assign Nouns to Entities in Dictionary**
  - Look for matching name
  - Search for synonym with matching name
- **Assign Actions to Verbs in Dictionary**
  - Look for matching name
  - Search for synonym with matching name
- **Identify Links to Existing Entities**
  - If entity with same name already exists, do not add entity but link to that entity
- **Identify/Check Relationships**
  - Identify actions without actors
  - Identify inputs without sender action
  - Identify outputs without receiver action
  - Identify action entity interactions without action, entity & role

Issues & Lessons Learned

ISSUES--REGISTER

- **Level or Granularity of Update/Register**
  - Cycle 1: Register/Update entire model
- **How frequently will programs update/create new versions?**
- **At level will programs create or update data?**

JSIMS CONVERSION ISSUES

Format: MS Word Documents

- **Content: Abundant. All information needed to support Cycle 1 (plus more) was available**
- **Some syntactical and semantical elements identifiable**
- **Inconsistencies in use of style guides made automated conversion impossible for some Cycle 1 fields (e.g., inputs, outputs, references)**
- **Data stored in header could not be read**
- **Need to increase length of some of CMMS fields (Action Names and Codes) to reflect JSIMS data**

WARSIM CONVERSION ISSUES

Format: Sample extract from UNISQL data base

- **Content: Limited or unclear. Information on entities and actions (tasks) was available; other Cycle 1 information not available in sample**
- **Format: Consistent and data base readable**
- **Sometimes difficult to separate out different syntactic and semantical elements**
- **FDB defines entities, attributes, and actions but no inputs and outputs**

NASM CONVERSION ISSUES

Format: RDD 100 RDT files

- **Content: Abundant. All information needed to support Cycle 1 (plus more) was available**
- **Format: Consistent and data base readable**
- **Syntactical and semantical elements identifiable**

JWARS CONVERSION ISSUES

Format: Statemate extract

- **Content: Abundant. Almost all information needed to support Cycle 1 was available**
- **More data could be converted in subsequent iterations by modifying collection procedures**
- **Format: Consistent and data base readable**
- **Syntactical and semantical elements identifiable**

GENERAL CONVERSION ISSUES

- **Knowledge should be captured in "data base readable" format**
  - Otherwise, conversion (& integration) will be difficult, costly, and inaccurate and DIF concept cannot be implemented
- **DIFs must be developed iteratively**
  - Need to work on agreement on content and format
  - Need agreement on common set of data dictionaries
- **Need technical POC with budgeted time to support conversion for each program**
- **Separate but related DIFs needed for exporting from CMMS to raw format (not considered during Cycle 1)**

DATA BASE READABLE

Format	***Rating***
Unstructured & Ungrouped Text files	***Worst***
Semi-structured and Semi-grouped text files	
ASCII Delimited, Fixed Width on consistently tagged text files	***Minimum Acceptable***
“Common” spreadsheet format (Excel, Lotus)	
RDBMS Standard Format (SQL, ODBC)	
Meta-standard format (e.g., Metadata Coalition)	
CMMS DBMS “mirror” image format	***Best***

PROPOSED DIF STRUCTURE

- **Describe Syntactical Structure & Links**
  - Describe source data base structure(files, record, relationships) via interchange standard (e.g., Metadata Interchange Format or MIF)
  - Describe linkages between source and CMMS DB via interchange standard (via MIF)

PROPOSED DIF STRUCTURE

- **Describe Semantics Elements and Links**
  - Provide program data dictionary
  - Describe links between source and CMMS data dictionaries
  - Describe recommended additions to CMMS data dictionaries

DICTIONARY AND SEMANTIC ISSUES

- **CMMS dictionaries must be developed ASAP and provided to programs prior to conversion**
- **Programs should (eventually) be responsible for linking to CMMS dictionaries**
- **Dictionaries--Entities, Verbs, Actions (Actor-Verb-Object), Attributes**

DICTIONARY AND SEMANTIC ISSUES (CONTINUED)

- **Verb Dictionary Recommendation**
  - Use single verb with single meaning
  - Put verb at beginning of action
  - Use first person present tense

- **Action Data Dictionary**
  - Action (Actor-Verb-Object)
  - Requires either separate storage of items or special tagging for automatic identification

- **Programs must only use items references in their dictionaries.**

OTHER INTEGRATION ISSUES

- **Warning vs Errors (What are required data elements?)**
- **Where should integration errors be corrected (original vs CMMS)**

LOCATE AND EXTRACT ISSUES

- What additional views, reports, & queries are needed?
- What is the priority for the additional reports?
- What is the best implementation design?
- Format for downloading data?

EVALUATE ISSUES

- Need policy defining CMMS verification, validation, certification, and accreditation and associated levels
- Is VV&A done prior to conversion and integration, after, or both?
- If we allow VV&A update in CMMS, how do we synchronize back to raw data?
- How are VV&A values aggregated?

MANAGE & RELEASE ISSUES

- **Level of user access**
  - Cycle 1: Data base structure designed to provide individuals access by role (e.g., examiner) and model
  - Dialogs tied to role
- **Storage of multiple versions of same model**
- **Internet security--MSRR solutions?**

ARCHITECTURAL ISSUES

- ODBC, ISAPI, Visual C++ DLLs and Internet Data Base Connector all worked well
- Front Page has some limitations (HTML editing, reformats pages)
- Access is very good for rapid prototyping but has performance/management problems for large data sets
- JAVA tool set was successfully employed. JAVA had some minor negative impacts on performance (speed), printing, and integration with other Netscape display features

CYCLE 2 OPTIONS

- Add functionality to capture additional elements in Technical Framework (e.g., conditions, sequencing)
- Add functionality to reflect new Technical Framework functions
- Develop DIFs and improved conversion routines for existing programs
- Develop RDBMS-based tool for capturing data In CMMS format
- Expand to additional areas (e.g., equipment, environment)